Podstawy języka R - zadania (3)

Tomasz Owczarek, Mateusz Naramski

2023/2024, semestr letni

Praca na danych - pasażerowie Titanika

31. Wykonaj poniższy kod, który pod zmienną `tt` wczyta dane z pliku *titanic.csv* i zamieni odpowiednie kolumny z tej ramki na zmienne kategoryczne (factory):

```
tt <- read.csv(file = "./data/titanic.csv")

tt$Survived <- factor(tt$Survived, levels = c(0, 1),
                      labels = c("not survived", "survived"))
tt$Pclass <- factor(tt$Pclass)
tt$Sex <- factor(tt$Sex)
tt$Embarked <- factor(tt$Embarked)
```

a następnie:

a) wyświetl liczbę rekordów tej ramki (*funkcja nrow*)
b) wyświetl liczbę pasażerów z podziałem na płeć (*funkcja table*)
c) średnią cenę biletu (kolumna `Fare`)
d) średnią cenę biletu kobiet (przyrównaj kolumnę `Sex` do *"female"* i podstaw pod kolumnę `Fare`)
e) liczbę pasażerów, którzy zapłacili za bilet więcej niż 100 dolarów.

```
## [1] 891
```

```
##
## female   male
##    314    577
```

```
## [1] 32.20421
```

```
## [1] 44.47982
```

```
## [1] 53
```

32. Pracując na ramce `tt` wyświetl:

a) liczbę pasażerów z poszczególnych klas (`Pclass`)
b) średnią cenę biletów z pierwszej klasy
c) średnią cenę biletów z każdej klasy (skorzystaj z funkcji `tapply`)
d) liczbę pasażerów z klasy 2, którzy zapłacili za bilet mniej niż 20 dolarów

```
##
##   1   2   3
## 216 184 491
```

```
## [1] 84.15469
```

```
##        1        2        3
## 84.15469 20.66218 13.67555
```

```
## [1] 104
```

33. Pracując na ramce `tt` wyświetl:

a) średni wiek pasażerów (w parametrze funkcji `mean` pomiń wartości `NA`)
b) min, max, średnią i kwartyle wieku pasażerów oraz ile wartości brakuje
c) średni wiek mężczyzn
d) średni wiek dla każdej płci (skorzystaj z funkcji `tapply` i konstrukcji funkcji anonimowej)

```
## [1] 29.69912
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.42   20.12   28.00   29.70   38.00   80.00     177
```

```
## [1] 30.72664
```

```
##   female     male
## 27.91571 30.72664
```

34. Pracując na ramce `tt` wyświetl:

a) liczbę osób każdej płci, o których nie ma informacji o ich wieku (skorzystaj z funkcji `is.na` z zdania 29.)
b) nazwiska, płeć, klasę i wiek osób mających przynajmniej 70 lat
c) maksymalny wiek w każdej klasie

```
##
## female   male
##     53    124
```

```
##                                      Name  Sex Pclass  Age
## 97             Goldschmidt, Mr. George B male      1 71.0
## 117                 Connors, Mr. Patrick male      3 70.5
## 494              Artagaveytia, Mr. Ramon male      1 71.0
## 631 Barkworth, Mr. Algernon Henry Wilson male      1 80.0
## 673          Mitchell, Mr. Henry Michael male      2 70.0
## 746         Crosby, Capt. Edward Gifford male      1 70.0
## 852                  Svensson, Mr. Johan male      3 74.0
```

```
##  1  2  3
## 80 70 74
```

35. Pracuj na ramce `tt`.

a) Wyświetl liczbę pasażerów z każdego portu.
b) Dwoje pasażerów nie ma informacji o porcie, z którego się okrętowali (ta wartość to nie `NA` tylko pusty ciąg znaków " "). Wyświetl ich imiona, płeć, wiek, klasę oraz kolumny Embarked i Cabin.

```
##
##       C   Q   S
##   2 168  77 644
```

```
##                                             Name    Sex Age Pclass Embarked Cabin
## 62                          Icard, Miss. Amelie female  38      1            B28
## 830 Stone, Mrs. George Nelson (Martha Evelyn) female  62      1            B28
```

36. Pracując na ramce `tt` wyświetl:

a) liczbę osób z podziałem na te, które przeżyły i nie przeżyły
b) procent osób z podziałem na te, które przeżyły i nie przeżyły
c) liczbę osób z każdej klasy z podziałem na te, które przeżyły i nie przeżyły
d) jw. ale w ujęciu procentowym

```
##
## not survived     survived
##          549          342
```

```
##
## not survived     survived
##     61.61616     38.38384
```

```
##
##                   1   2   3
##   not survived  80  97 372
##   survived     136  87 119
```

```
##
##                         1         2         3
##   not survived  8.978676 10.886644 41.750842
##   survived     15.263749  9.764310 13.355780
```

37. Pracując na ramce `tt` wyświetl:

a) dla osób, które przeżyły i nie przeżyły rozkład procentowy klas, którymi podróżowali
b) dla osób z każdej klasy procent tych, którzy przeżyli nie przeżyli

```
##
##                       1        2        3
##   not survived 14.57195 17.66849 67.75956
##   survived     39.76608 25.43860 34.79532
```

```
##
##                       1        2        3
##   not survived 37.03704 52.71739 75.76375
##   survived     62.96296 47.28261 24.23625
```

38. Sprawdź, jaki procent osób przeżył i nie przeżył wśród tych osób, o których mamy wszystkie informacje oraz tych, których rekordy mają braki (skorzystaj z funkcji `complete.cases`).

```
##
##                    FALSE     TRUE
##   not survived 70.62147 59.38375
##   survived     29.37853 40.61625
```

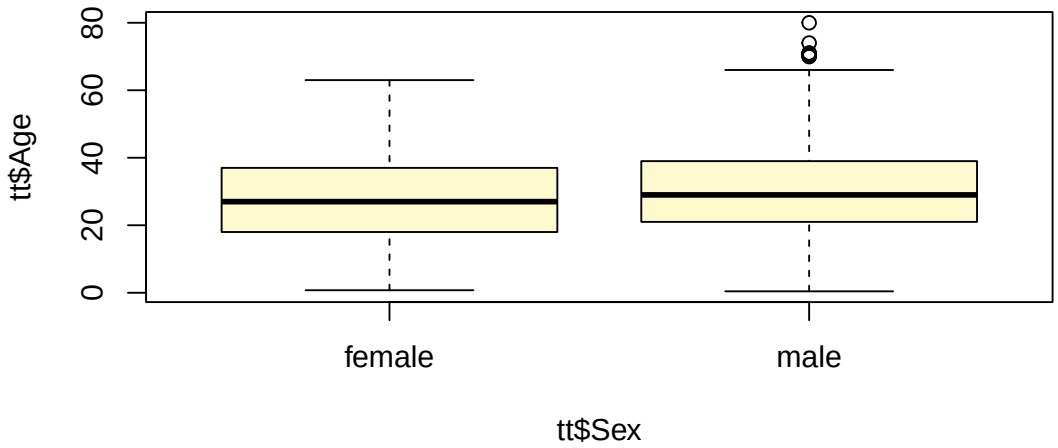
39. Dołóż do danych tt nową kolumnę o nazwie `Relatives`, która dla każdego pasażera będzie zawierała łączną liczbą wszystkich krewnych na pokładzie (czyli sumę kolumn `Sibsp` i `Parch`). Następnie korzystając z tej kolumny sprawdź:

a) ile osób podróżowało bez krewnych
b) jaki procent osób podróżował z poszczególną liczbą krewnych (wynik zaokrąglij do dwóch miejsc po przecinku)

```
## [1] 537
```

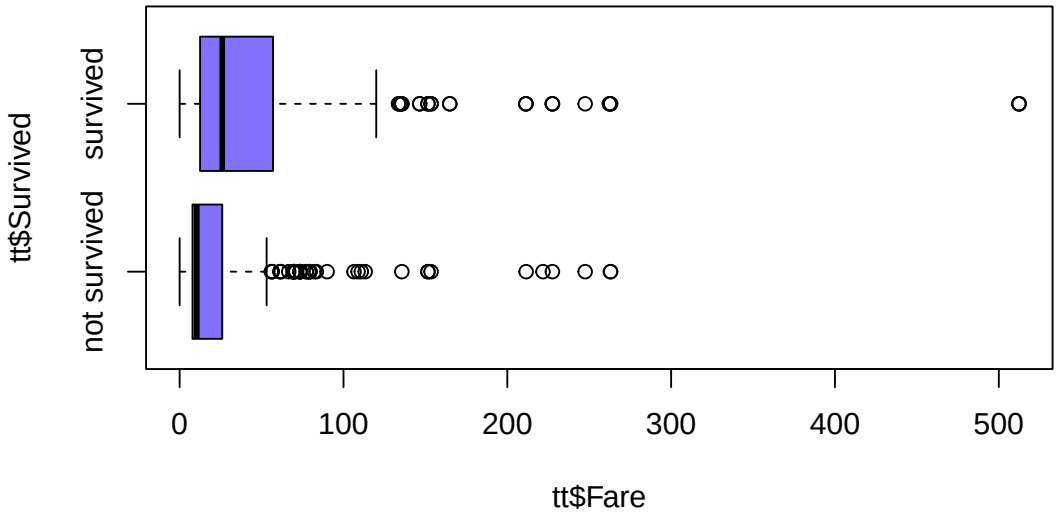
```
##
##     0     1     2     3     4     5     6     7    10
## 60.27 18.07 11.45  3.25  1.68  2.47  1.35  0.67  0.79
```

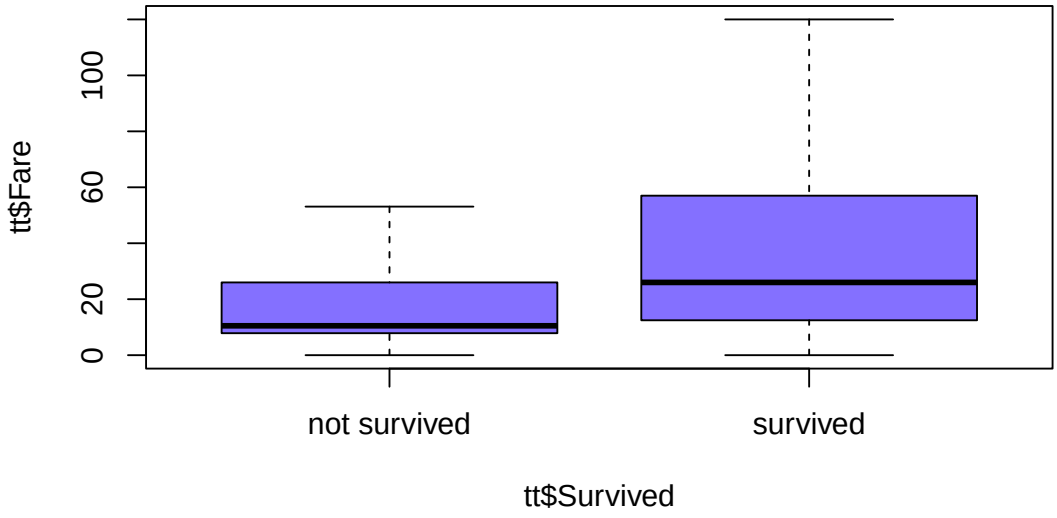
40. Pracuj na ramce `tt`. Wyświetl wykres pudełkowy wieku osób z podziałem na płeć. Pudełka mają mieć kolor o ciekawej nazwie, np. *lemonchiffon*.



41. Pracuj na ramce `tt`.

a) Wyświetl poziome wykresy pudełkowe cen biletów z podziałem na osoby, które przeżyły i nie przeżyły. Pudełka mają mieć kolor o ciekawej nazwie, ale inny niż w zadaniu 40., np. *lightslateblue*.
b) Wyświetl to, co w punkcie a), ale pionowo i bez wartości odstających (*sprawdź w pomocy funkcji boxplot, jak nie wyświetlać wartości odstających, czyli tzw. outliers*).





42. Pracuj na ramce `tt`.

a) Wyświetl w postaci wektora wiek 20 najstarszych osób posortowany malejąco
b) Wyświetl imiona, wiek, płeć oraz informacje o przeżyciu 10 najmłodszych osób posortowanych rosnąco wg wieku

```
##  [1] 80.0 74.0 71.0 71.0 70.5 70.0 70.0 66.0 65.0 65.0 65.0 64.0 64.0 63.0 63.0
## [16] 62.0 62.0 62.0 62.0 61.0
```

```
##                                Name  Age    Sex     Survived
## 804 Thomas, Master. Assad Alexander 0.42   male     survived
## 756       Hamalainen, Master. Viljo 0.67   male     survived
## 470   Baclini, Miss. Helene Barbara 0.75 female     survived
## 645          Baclini, Miss. Eugenie 0.75 female     survived
## 79    Caldwell, Master. Alden Gates 0.83   male     survived
## 832 Richards, Master. George Sibley 0.83   male     survived
## 306  Allison, Master. Hudson Trevor 0.92   male     survived
## 165    Panula, Master. Eino Viljami 1.00   male not survived
## 173    Johnson, Miss. Eleanor Ileen 1.00 female     survived
## 184       Becker, Master. Richard F 1.00   male     survived
```

43. Pracuj na ramce `tt`.

a) Sprawdź, ile osób ma w kolumnie `Name` ciąg znaków "Ann".
b) Wyświetl te imiona w postaci wektora.

```
## [1] 31
```

```
##  [1] "McGowan, Miss. Anna \"Annie\""
##  [2] "Turpin, Mrs. William John Robert (Dorothy Ann Wonnacott)"
##  [3] "Laroche, Miss. Simonne Marie Anne Andree"
##  [4] "Faunthorpe, Mrs. Lizzie (Elizabeth Anne Wilkinson)"
##  [5] "Salkjelsvik, Miss. Anna Kristine"
##  [6] "Andersson, Miss. Ellis Anna Maria"
##  [7] "Peter, Miss. Anna"
##  [8] "Nysten, Miss. Anna Sofia"
##  [9] "Skoog, Mrs. William (Anna Bernhardina Karlsson)"
## [10] "Isham, Miss. Ann Elizabeth"
## [11] "Cameron, Miss. Clear Annie"
## [12] "Hamalainen, Mrs. William (Anna)"
## [13] "Ward, Miss. Anna"
## [14] "Mellinger, Mrs. (Elizabeth Anne Maidment)"
## [15] "Kelly, Miss. Anna Katherine \"Annie Kate\""
## [16] "Lahtinen, Mrs. William (Anna Sylfven)"
## [17] "Funk, Miss. Annie Clemmer"
## [18] "Warren, Mrs. Frank Manley (Anna Sophia Atkinson)"
## [19] "Jermyn, Miss. Annie"
## [20] "Meek, Mrs. Thomas (Annie Louise Rowley)"
## [21] "Danbom, Mrs. Ernst Gilbert (Anna Sigrid Maria Brogren)"
## [22] "Hoyt, Mrs. Frederick Maxfield (Jane Anne Forby)"
## [23] "Perreault, Miss. Anne"
## [24] "Turja, Miss. Anna Sofia"
## [25] "Harper, Miss. Annie Jessie \"Nina\""
## [26] "Ford, Mrs. Edward (Margaret Ann Watson)"
## [27] "Sinkkonen, Miss. Anna"
## [28] "Hogeboom, Mrs. John C (Anna Andrews)"
## [29] "Sjoblom, Miss. Anna Sofia"
## [30] "Sage, Miss. Stella Anna"
## [31] "Collyer, Mrs. Harvey (Charlotte Annie Tate)"
```

44. Pracuj na ramce `tt`.

a) Wyświetl średni wiek osób, którzy w imieniu mają ciąg znaków "Master"
b) Wyświetl średni wiek osób, którzy w imieniu mają ciąg znaków "Mr." (skorzystaj z opcji *fixed* w funkcji `grepl`)
c) Wśród pasażerów było 6 pastorów. Można ich zidentyfikować po tytule "Rev.". Wyświetl ich nazwiska, wiek, klasę oraz informację, czy przeżyli.

```
## [1] 4.574167
```

```
## [1] 32.36809
```

```
##                                 Name Age Pclass     Survived
## 150 Byles, Rev. Thomas Roussel Davids  42      2 not survived
## 151        Bateman, Rev. Robert James  51      2 not survived
## 250     Carter, Rev. Ernest Courtenay  54      2 not survived
## 627    Kirkland, Rev. Charles Leonard  57      2 not survived
## 849                Harper, Rev. John  28      2 not survived
## 887            Montvila, Rev. Juozas  27      2 not survived
```

Zadania różne

45. Wczytaj dane z pliku *movies.csv* pod zmienną o nazwie `mov` (użyj do tego funkcji `read.csv2`, otwórz dane w notatniku i sprawdź dlaczego). Następnie zamień kolumnę `genre` na zmienną kategoryczną (*factor*).

a) Sprawdź strukturę tej ramki
b) Zrób podsumowanie wszystkich danych w ramce

```
## 'data.frame':    1565 obs. of  11 variables:
## $ title : chr "Glitter" "Soul Survivors" "Megiddo: The Omega Code 2" "On the
##    Line" ...
## $ genre : Factor w/ 4 levels "Action","Adventure",..: 4 4 1 3 1 1 3 3 3 3 ...
## $ director : chr "Vondie Curtis-Hall" "Stephen Carpenter" "Brian
##    Trenchard-Smith" "Eric Bross" ...
## $ year : int 2001 2001 2001 2001 2001 2001 2001 2001 2001 2001 ...
## $ duration : int 104 84 104 85 85 116 87 93 86 87 ...
## $ gross : int 4273372 3100650 5974653 4356743 12610731 32616869 14249005
##    10097096 23978402 112950721 ...
## $ budget : int 22000000 14000000 22000000 16000000 11000000 72000000 15000000
##    4000000 11000000 72000000 ...
## $ cast_facebook_likes: int 1854 417 4221 2446 3050 14780 2689 955 3850 3287 ...
## $ votes : int 19412 7277 2253 3662 38985 34435 39788 5612 10966 33884 ...
## $ reviews : int 374 245 129 125 878 455 716 65 162 170 ...
## $ rating : num 2.1 3.9 4.1 4.1 4.4 4.5 4.5 4.6 4.6 4.6 ...
```

```
##     title                genre       director             year
##  Length:1565        Action   :471   Length:1565        Min.   :2001
##  Class :character   Adventure:194   Class :character   1st Qu.:2004
##  Mode  :character   Comedy   :570   Mode  :character   Median :2008
##                     Drama    :330                      Mean   :2008
##                                                        3rd Qu.:2012
##                                                        Max.   :2016
##     duration         gross              budget          cast_facebook_likes
##  Min.   : 63.0   Min.   :      703   Min.   :     7000   Min.   :     0
##  1st Qu.: 95.0   1st Qu.: 12870569   1st Qu.: 15000000   1st Qu.:  2579
##  Median :104.0   Median : 36931089   Median : 30000000   Median :  5707
##  Mean   :107.5   Mean   : 63245128   Mean   : 49589620   Mean   : 14001
##  3rd Qu.:117.0   3rd Qu.: 81257845   3rd Qu.: 68000000   3rd Qu.: 19359
##  Max.   :280.0   Max.   :760505847   Max.   :300000000   Max.   :656730
##      votes            reviews          rating
##  Min.   :     91   Min.   :   6.0   Min.   :1.900
##  1st Qu.:  23671   1st Qu.: 236.0   1st Qu.:5.700
##  Median :  60156   Median : 391.0   Median :6.400
##  Mean   : 113832   Mean   : 548.9   Mean   :6.253
##  3rd Qu.: 146352   3rd Qu.: 685.0   3rd Qu.:7.000
##  Max.   :1676169   Max.   :5312.0   Max.   :9.000
```

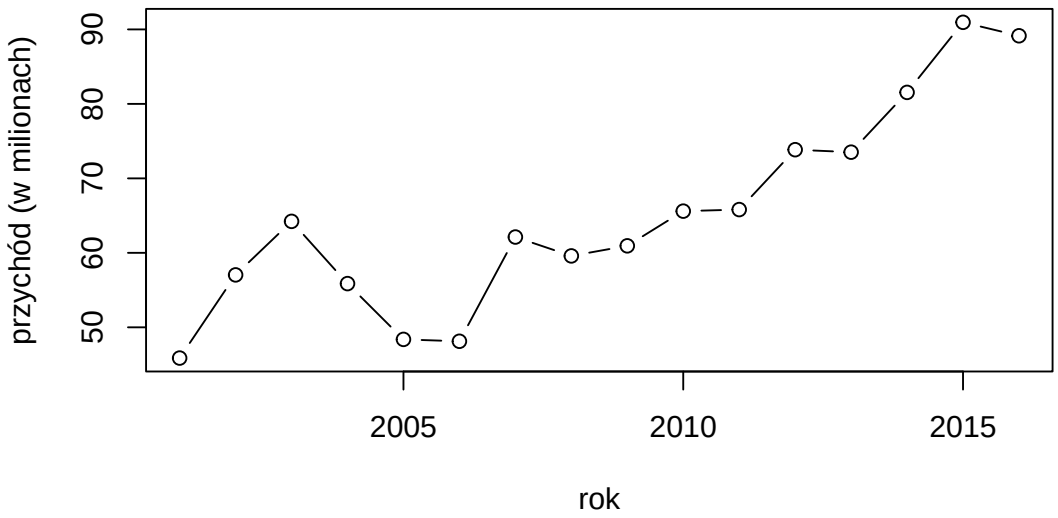
46. Pracuj na ramce `mov`.

a) Sprawdź liczbę filmów z poszczególnych lat.
b) Sprawdź średni przychód (*gross*) filmów z poszczególnych lat.

```
##
## 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016
##  111  115  102  106  108  104   79  115  109  109  106  104   96   91   72   38
```

```
##      2001      2002      2003      2004      2005      2006      2007      2008
## 45872182 57038748 64226758 55870123 48378666 48124718 62123002 59587575
##      2009      2010      2011      2012      2013      2014      2015      2016
## 60942150 65600007 65808707 73851353 73516734 81545904 90953540 89145848
```

47. Przedstaw średni przychód filmów z poszczególnych lat (w milionach) na wykresie.



48. Wykonaj poniższy kod. Spowoduje on wczytanie pod zmienną `pokemon1_4` danych z pliku *pokemon1_4.csv*, a pod zmienną `pokemon5_6` danych z pliku *pokemon5_6.csv*.

```
pokemon1_4 <- read.csv("./data/pokemon1_4.csv", skip = 3)
pokemon5_6 <- read.csv("./data/pokemon5_6.csv")
```

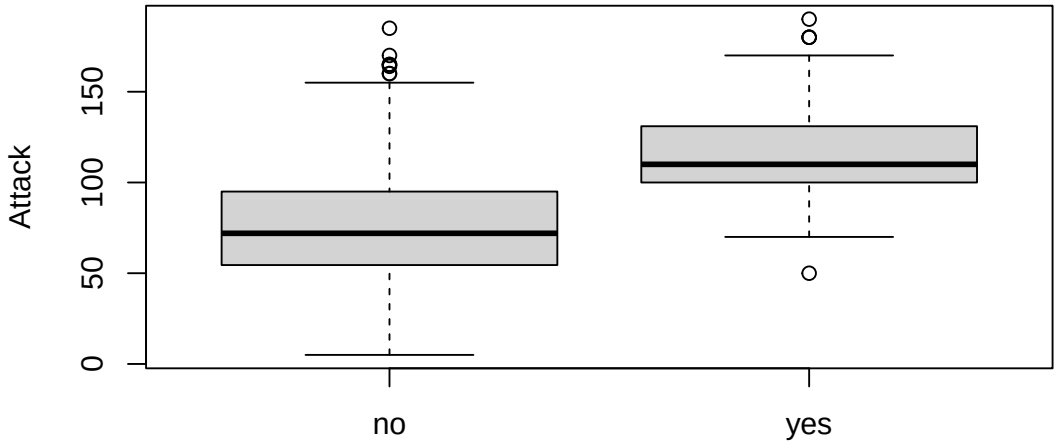
a) Dlaczego w pierwszym przypadku wewnątrz `read.csv` podano argument `skip`? (*sprawdź, jak wygląda zawartość pliku pokemon1_4.csv*)
b) Korzystając z funkcji `rbind` (przeczytaj jej dokumentację) utwórz nową zmienną o nazwie `pokemon`, która będzie zawierała dane o wszystkich 800 pokemonach
c) Pierwsza kolumna w ramce `pokemon` o nazwie *X.* jest niepotrzebna. Usuń ją (*pod zmienną* `pokemon` *podstaw ramkę bez pierwszej kolumny*)
d) Zmień nazwy kolumn *Type.1*, *Type.2*, *Sp..Atk*, *Sp..Def* na *Type1*, *Type2*, *SpAtk*, *SpDef* (*skorzystaj z funkcji* `names`)
e) Zamień kolumnę *Legendary* na zmienna kategoryczną z wartościami "no" i "yes"

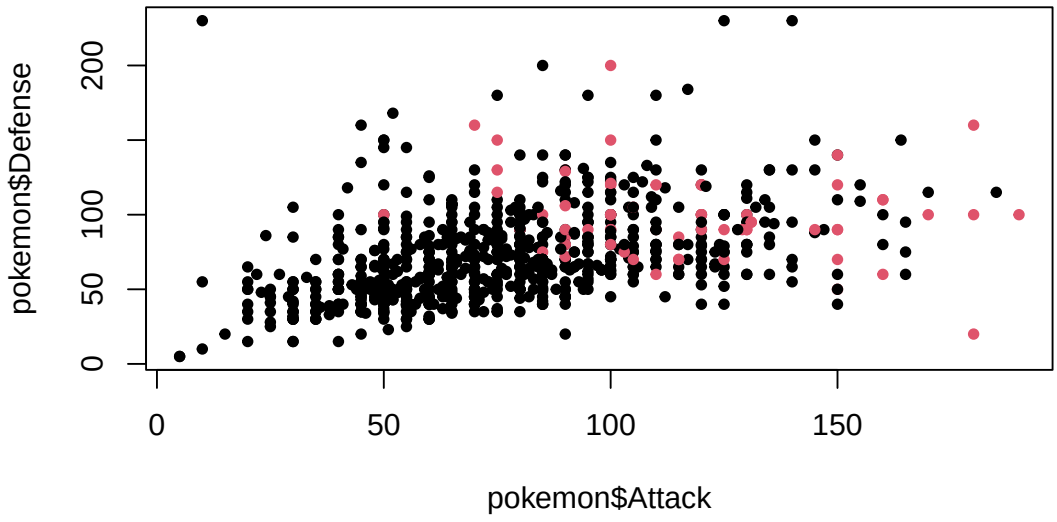
Wykonanie polecenia `str(pokemon)` powinno zwrócić po tym następujący wynik:

```
## 'data.frame':    800 obs. of  12 variables:
## $ Name : chr "Bulbasaur" "Ivysaur" "Venusaur" "VenusaurMega Venusaur" ...
## $ Type1 : chr "Grass" "Grass" "Grass" "Grass" ...
## $ Type2 : chr "Poison" "Poison" "Poison" "Poison" ...
## $ Total : int 318 405 525 625 309 405 534 634 634 314 ...
## $ HP : int 45 60 80 80 39 58 78 78 78 44 ...
## $ Attack : int 49 62 82 100 52 64 84 130 104 48 ...
## $ Defense : int 49 63 83 123 43 58 78 111 78 65 ...
## $ SpAtk : int 65 80 100 122 60 80 109 130 159 50 ...
## $ SpDef : int 65 80 100 120 50 65 85 85 115 64 ...
## $ Speed : int 45 60 80 80 65 80 100 100 100 43 ...
## $ Generation: int 1 1 1 1 1 1 1 1 1 1 ...
## $ Legendary : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
```

49. Korzystając z danych `pokemon`:

a) utwórz wykres pudełkowy pokazujący rozkład cechy Attack z podziałem na legendarne i nielegendarne pokemony (zwróć uwagę na podpisy osi)
b) utwórz wykres punktowy przedstawiający zależność między atakiem a obroną (użyj koloru, żeby rozróżnić legendarne i nielegendarne pokemony, wewnątrz funkcji `plot` ustaw parametr *pch* na 19, żeby zmienić domyślny kształt punktów)





Zadanie dodatkowe

50! Czasami dane nie są zapisane jako kolumny rozdzielone separatorami, ale w inny sposób (szczególnie wtedy, gdy jest dużo kolumn takiego samego typu). Przykładem takich danych jest plik scores.txt. Wczytaj i ewentualnie przekształć te dane do takiej postaci, żeby wykonanie polecenia `str()` na tych danych zwróciło poniższy wynik. Cały kod powinien liczyć nie więcej niż 200 znaków (uwaga - nazwy pliku z danymi nie wolno zmieniać).

```
## 'data.frame':    43 obs. of  12 variables:
##  $ id     : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ sex    : chr  "M" "F" "F" "M" ...
##  $ score1 : int  1 1 1 0 1 1 1 2 1 1 ...
##  $ score2 : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ score3 : int  3 2 1 1 3 2 1 2 3 2 ...
##  $ score4 : int  3 2 2 3 3 2 2 0 3 2 ...
##  $ score5 : int  1 3 2 0 1 3 2 0 1 3 ...
##  $ score6 : int  1 1 1 2 1 1 1 2 1 1 ...
##  $ score7 : int  1 1 1 0 1 1 1 0 1 1 ...
##  $ score8 : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ score9 : int  3 2 1 1 3 2 1 2 3 2 ...
##  $ score10: int  3 2 2 3 3 2 2 0 3 2 ...
```